UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P O Box 1450
Alexandria, Virginia 22313-1450
www uspto gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/005,248	12/03/2001	Brian C. Barnes	2000.056500	7937

23720 7590 11/06/2009
WILLIAMS, MORGAN & AMERSON
10333 RICHMOND, SUITE 1100
HOUSTON, TX 77042

EXAMINER	
LEMMA, SAMSON B	
ART UNIT	PAPER NUMBER
2432	
MAIL DATE	DELIVERY MODE
11/06/2009	PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE BOARD OF PATENT APPEALS
AND INTERFERENCES

Ex parte BRIAN C. BARNES, RODNEY W. SCHMIDT, and
GEOFFREY S. STRONGIN

Appeal 2008-006005
Application 10/005,248
Technology Center 2400

Decided: November 6, 2009

Before LEE E. BARRETT, LANCE LEONARD BARRY, and
JOHN A. JEFFERY, *Administrative Patent Judges*.

BARRY, *Administrative Patent Judge*.

DECISION ON APPEAL

STATEMENT OF THE CASE

The Patent Examiner rejected claims 1-24. The Appellants appeal therefrom under 35 U.S.C. § 134(a). We have jurisdiction under 35 U.S.C. § 6(b).

INVENTION

The invention at issue on appeal "restrict[s] the execution of security sensitive instructions." (Spec. 22.)

ILLUSTRATIVE CLAIM

1. A method, comprising:

> associating a first security identification (ID) with each of a plurality of instructions or a set of instructions that are to be executed by a processor;
>
> requesting to execute at least one of the plurality of instructions or set of instructions by a software code running on the processor;
>
> obtaining a second security ID associated with the software code running on the processor;
>
> comparing the second security ID with the first security ID; and
>
> executing the requested instruction or set of instructions providing that the second security ID matches the first security ID.

PRIOR ART

Kamiya	4,949,238	Aug. 14, 1990
Krueger	4,962,533	Oct. 9, 1990
Draves	5,802,590	Sept. 1, 1998

REJECTIONS

Claims 1-3, 7-11, 15-19, 23, and 24 stand rejected under 35 U.S.C. § 102(b) as being anticipated by Draves.

Claims 4-6, 12-14, and 20-22 stand rejected under 35 U.S.C. § 103(a) as being unpatentable over Draves and Krueger.

Claims 1-3, 7-11, 15-19, 23, and 24 stand rejected under 35 U.S.C. § 102(b) as being anticipated by Kamiya.

Claims 4-6, 12-14, and 20-22 stand rejected under 35 U.S.C. § 103(a) as being unpatentable over Kamiya and Krueger.

ANTICIPATION AND OBVIOUSNESS REJECTIONS RELYING ON DRAVES

The Examiner makes the following finding about Draves.

> [I]t is [sic] been disclosed on column 1, lines 35-38, the kernel is requested so that the word processing program which meets the limitation of software code shares a block of memory with the Spread sheet program/plurality of instructions or set of instruction so that the word processing program/software code can access the spreadsheet data.

(Ans. 18.) The Appellants argue that "none of the various resources described by Draves include at least one of the plurality of instructions or set of instructions that have been requested to execute by the software code, as set forth in independent claims 1, 9, and 17." (App. Br. 15.)

ISSUE

Therefore, the issue before us is whether the Appellants have shown error in the Examiner's finding that Draves associates a security ID with at least one executable instruction that a software code has requested to execute.

LAW

"It is axiomatic that anticipation of a claim under § 102 can be found only if the prior art reference discloses every element of the claim, and that anticipation is a fact question . . . ." *In re King*, 801 F.2d 1324, 1326 (Fed. Cir. 1986) (citing *Lindemann Maschinenfabrik GMBH v. Am. Hoist & Derrick Co.,* 730 F.2d 1452, 1457 (Fed. Cir. 1984)).

FINDINGS OF FACT (FFS)

1. Draves includes the following teachings.

> Processes frequently need to share resources. For example, a spreadsheet program may need to pass spreadsheet data to a word processing program so that the word processing program can incorporate the spreadsheet data into a document. The spreadsheet program can pass the data to the word processing program using shared memory.

(Col. 1, ll. 22-29.)

2. The reference adds the following teachings.

> To pass the data, the spreadsheet program requests the kernel to allocate a block of memory for use as shared memory. In response, the kernel allocates the memory, maps the block of memory into the address space of the spreadsheet program, and returns to the spreadsheet program a resource identifier that identifies the block of memory. The spreadsheet program then writes the spreadsheet data to the block of memory and requests the kernel to allow the word processing program to share the block of memory. The kernel generates another resource identifier that the word processing program can use to access the block of memory. Once the block of memory is mapped into its address space, the word processing program can access the spreadsheet data.

(*Id.* ll. 27-41.)

3. Draves also includes the following description.

> Conventional computer systems are often controlled by multi-processing operating systems. These operating systems manage the concurrent execution of various computer programs. Each concurrently executing computer program is referred to as a process. These operating systems also control access to various resources of a computer system. These resources include the central processing unit, main memory, and peripheral devices (e.g., disk drives and printers).

(*Id.* ll. 11-19.)

ANALYSIS

Processes in Draves often share resources. (FF 1.) To do so, a kernel associates resource identifiers with the resource to be shared. (FF 2.) The resources shared by the processes, however, are not executable instructions. To the contrary, the Examiner admits that resource shares "a block of memory" (Ans. 18) or "spreadsheet data" (*id.*). The reference also discloses that resources include a central processing unit, main memory, and peripheral devices (e.g., disk drives and printers) (FF 3); none of which are executable instructions. Furthermore, the Examiner does not allege, let alone show, that the addition of Krueger cures the aforementioned deficiency of Draves.

CONCLUSION

Based on the aforementioned facts and analysis, we conclude that the Appellants have shown error in the Examiner's finding that Draves associates a security ID with at least one executable instruction that a software code has requested to execute.

ANTICIPATION REJECTION RELYING ON KAMIYA

37 C.F.R. § 41.37(c)(1)(vii) follows in pertinent part.

> When multiple claims subject to the same ground of rejection are argued as a group by appellant, the Board may select a single claim from the group of claims that are argued together to decide the appeal with respect to the group of claims as to the ground of rejection on the basis of the selected claim alone. Notwithstanding any other provision of this paragraph, the failure of appellant to separately argue claims which appellant has grouped together shall constitute a waiver of any argument that the Board must consider the patentability of any grouped claim separately.

Here, the Appellants argue claims 1-3, 7-11, 15-19, 23, and 24, which are subject to the same ground of rejection, as a group. (App. Br. 18-19.)

Therefore, we select claim 1 as the sole claim on which to decide the appeal of the group.

ISSUE

The Examiner finds that Kamiya discloses all the limitations of claim 1. (Ans. 9.) The Appellants argue that "*Kamiya* is completely silent with regard to requesting to execute at least one instruction by the software code running on the processor and executing the requested instruction." (App. Br. 19.) Therefore, the issue before us is whether the Appellants have shown error in the Examiner's finding that software code running on a processor of Kamiya requests execution of at least one instruction and subsequently executes the instruction.

LAW

"'On appeal to the Board, an applicant can overcome a rejection by showing insufficient evidence of prima facie obviousness or by rebutting the prima facie case with evidence of secondary indicia of nonobviousness.'" *In re Kahn*, 441 F.3d 977, 985-86 (Fed.Cir. 2006) (quoting *In re Rouffet*, 149 F.3d 1350, 1355 (Fed.Cir. 1998)).

FINDINGS OF FACT

4. The Examiner makes the following findings:

> Kamiya discloses a method, comprising
>
> . . . .
>
> - Requesting to execute at least one of the plurality of instructions of set of instructions by a

> software code running on the processor; [Column 5, lines 23-25) (branch instruction executed ... ...)
>
> . . . .
>
> - Executing the requested instruction or set of instructions [column 2, lines 22-24, "the succeeding microinstruction is normally selected"] . . . .

(Ans. 9.)

ANALYSIS

The Examiner makes specific finding about where Kamiya discloses the limitations at issue, mapping the exact limitations to specific parts of the reference and explaining the anticipation thereof. (FF 4.) Apart from merely summarily alleging that Kamiya does not disclose the recited limitations, the Appellants do not address, let alone show error in, the Examiner's specific findings.

CONCLUSION

Based on the aforementioned facts and analysis, we conclude that the Appellants have shown no error in the Examiner's finding that software code running on a processor of Kamiya requests execution of at least one instruction and subsequently executes the instruction.

OBVIOUSNESS REJECTION RELYING ON KAMIYA AND KRUEGER

The Appellants argue claims 4-6, 12-14, and 20-22, which are subject to the same ground of rejection, as a group. (App. Br. 19-21.) Therefore, we select claim 4 as the sole claim on which to decide the appeal of the group.

ISSUE

The Examiner finds that the combined teachings of Kamiya and Krueger would have suggested all the limitations of claim 4. (Ans. 10-11.) The Appellants argue that "Krueger is completely silent with regard to classification of an instruction accessing a word in the memory." (App. Br. 21.) Therefore, the issue before us is whether the Appellants have shown error in the Examiner's finding that the combined teachings of Kamiya and Krueger would have suggested classifying an executable instruction accessing a word in a memory.

LAW

"The test for obviousness is what the combined teachings of the references would have suggested to one of ordinary skill in the art." *In re Young*, 927 F.2d 588, 591 (Fed. Cir. 1991) (citing *In re Keller*, 642 F.2d 413, 425 (CCPA 1981)). "Non-obviousness cannot be established by attacking references individually where the rejection is based upon the teachings of a combination of references." *In re Merck & Co*., 800 F.2d 1091, 1097 (Fed. Cir. 1986) (citing *Keller*, 642 F.2d at 425). In determining obviousness, furthermore, a reference "must be read, not in isolation, but for what it fairly teaches in combination with the prior art as a whole." *Id.*

FINDINGS OF FACT

5. In Kruger's "computer system every word in the memory has a corresponding label. This label indicates the security classification, and compartments if any, of that word of data. Each time a word is accessed by any instruction, its classification is checked to see if access is allowed." (Col. 2, ll. 43-48.)

6. Kamiya discloses "an apparatus for detecting memory protection violations in microprogram controlled data processors." (Col. 1, ll. 7-9.)

7. Citing Kamiya, column 3, lines 25-27 (App. Br. 18), the Appellants admit that "[t]o detect a memory protection violation in a data processor for executing microinstructions under control of microprograms, [Kamiya's] apparatus comprises privilege level register means for storing a privilege level of a program now being executed" (*id.*). The part of the reference cited by the Appellants more specifically describes "a current privilege level register (CPL) 17 for storing a privilege level of a program now being executed . . . ." (Kamiya, col. 3, ll. 25-27.)

ANALYSIS

We agree with the Appellants' aforementioned argument that "Krueger is completely silent with regard to classification of an instruction accessing a word in the memory." (App, Br. 21.) Instead of classifying executable instructions, the reference classifies data. (FF 5.)

We also agree with the Appellants that Kamiya discloses a register for storing a privilege level of a program now being executed. (FF 7.) We find, however, that such storing a privilege level of a program being executed would have suggested classifying an executable instruction accessing a word in a memory.

CONCLUSION

Based on the aforementioned facts and analysis, we conclude that the Appellants have shown no error in the Examiner's finding that the combined teachings of Kamiya and Krueger would have suggested classifying an executable instruction accessing a word in a memory.

DECISION

We reverse the rejection of claims 1-3, 7-11, 15-19, 23, and 24 as being anticipated by Draves, and the rejection of claims 4-6, 12-14, and 20-22 under 35 U.S.C. § 103(a) as being unpatentable over Draves and Krueger. In contrast, we affirm the rejection of claims 1-3, 7-11, 15-19, 23, and 24 as being anticipated by Kamiya and the rejection of claims 4-6, 12-14, and 20-22 as being unpatentable over Kamiya and Krueger.

No time for taking any action connected with this appeal may be extended under 37 C.F.R. § 1.136(a)(1). *See* 37 C.F.R. § 1.136(a)(1)(iv).

AFFIRMED

llw

WILLIAMS, MORGAN & AMERSON
10333 RICHMOND, SUITE 1100
HOUSTON, TX 77042